

Assignment: Week 3 Practice Assignments (Individual)

Ex. 5-1 Enhance the Invoice Total application (Windows Forms project) (#1-7)

Open the application and change the if-else statement (#'s1-4)

The screenshot shows the Visual Studio IDE with the C# code for the InvoiceTotal application. The code is in the Form1.cs file, and the event handler for the btnCalculate\_Click method is shown. The code calculates the discount amount based on the subtotal and customer type. The code is as follows:

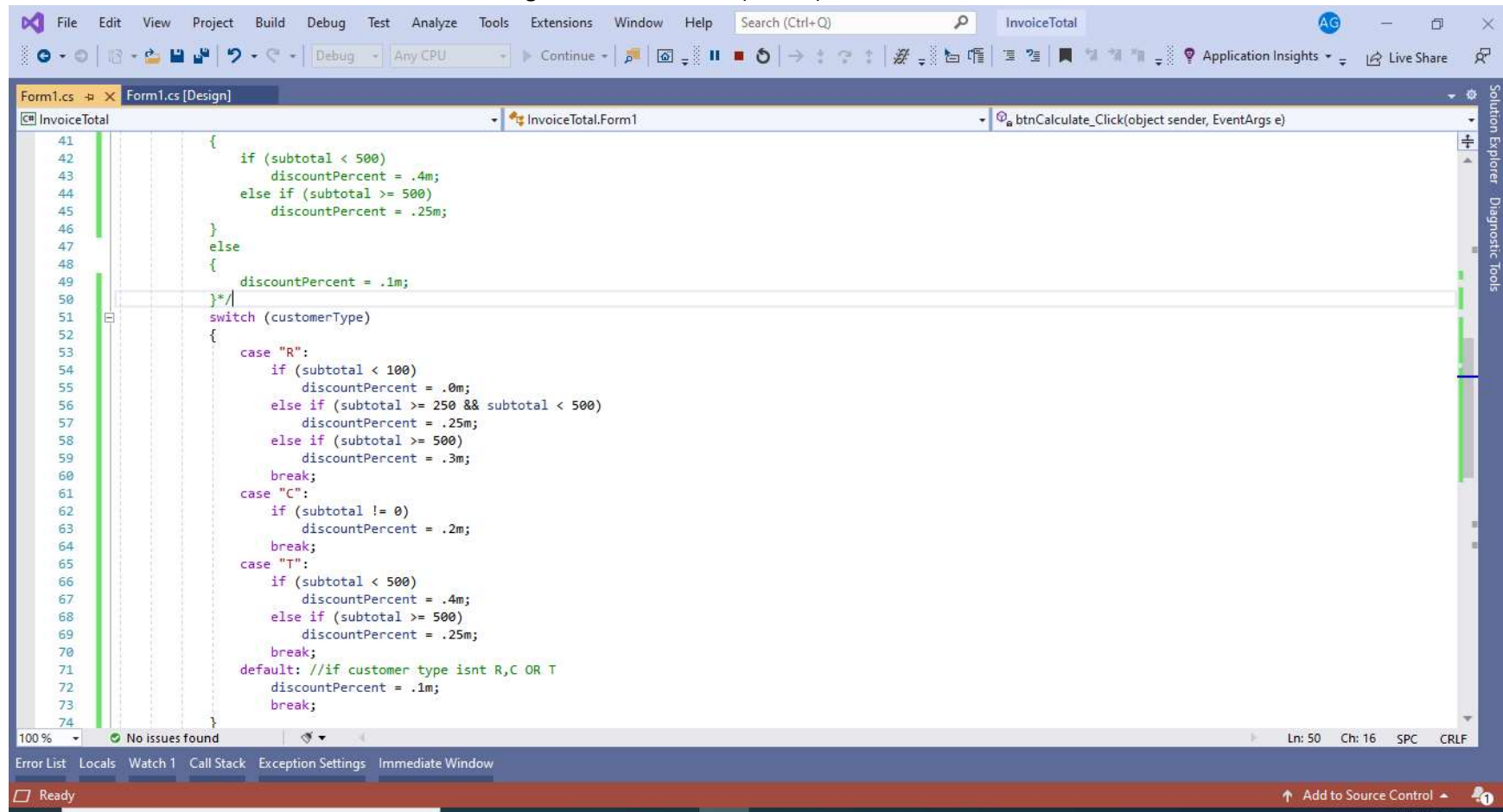
```
23 decimal subtotal = Convert.ToDecimal(txtSubtotal.Text);
24 decimal discountPercent = .0m;
25
26 if (customerType == "R")
27 {
28     if (subtotal < 100)
29         discountPercent = .0m;
30     else if (subtotal >= 250 && subtotal < 500)
31         discountPercent = .25m;
32     else if (subtotal >= 500)
33         discountPercent = .3m;
34 }
35 else if (customerType == "C")
36 {
37     if (subtotal != 0)
38         discountPercent = .2m;
39 }
40 else if (customerType == "T")
41 {
42     if (subtotal < 500)
43         discountPercent = .4m;
44     else if (subtotal >= 500)
45         discountPercent = .25m;
46 }
47 else
48 {
49     discountPercent = .1m;
50 }
51
52 decimal discountAmount = subtotal * discountPercent;
53 decimal invoiceTotal = subtotal - discountAmount;
54
55 txtDiscountPercent.Text = discountPercent.ToString("p1");
56 txtDiscountAmount.Text = discountAmount.ToString("c");
```

The Solution Explorer shows the project structure for InvoiceTotal, including Form1.cs, Form1.Designer.cs, Form1.resx, and Program.cs.

Below the code, four screenshots of the InvoiceTotal application are shown, illustrating the results of different inputs:

Customer type	Subtotal	Discount percent	Discount amount	Total
R	300	25.0%	\$75.00	\$225.00
R	600	30.0%	\$180.00	\$420.00
T	300	40.0%	\$120.00	\$180.00
F	300	10.0%	\$30.00	\$270.00

Use a switch statement with if-else statements to get the same results (#'s 5-7)

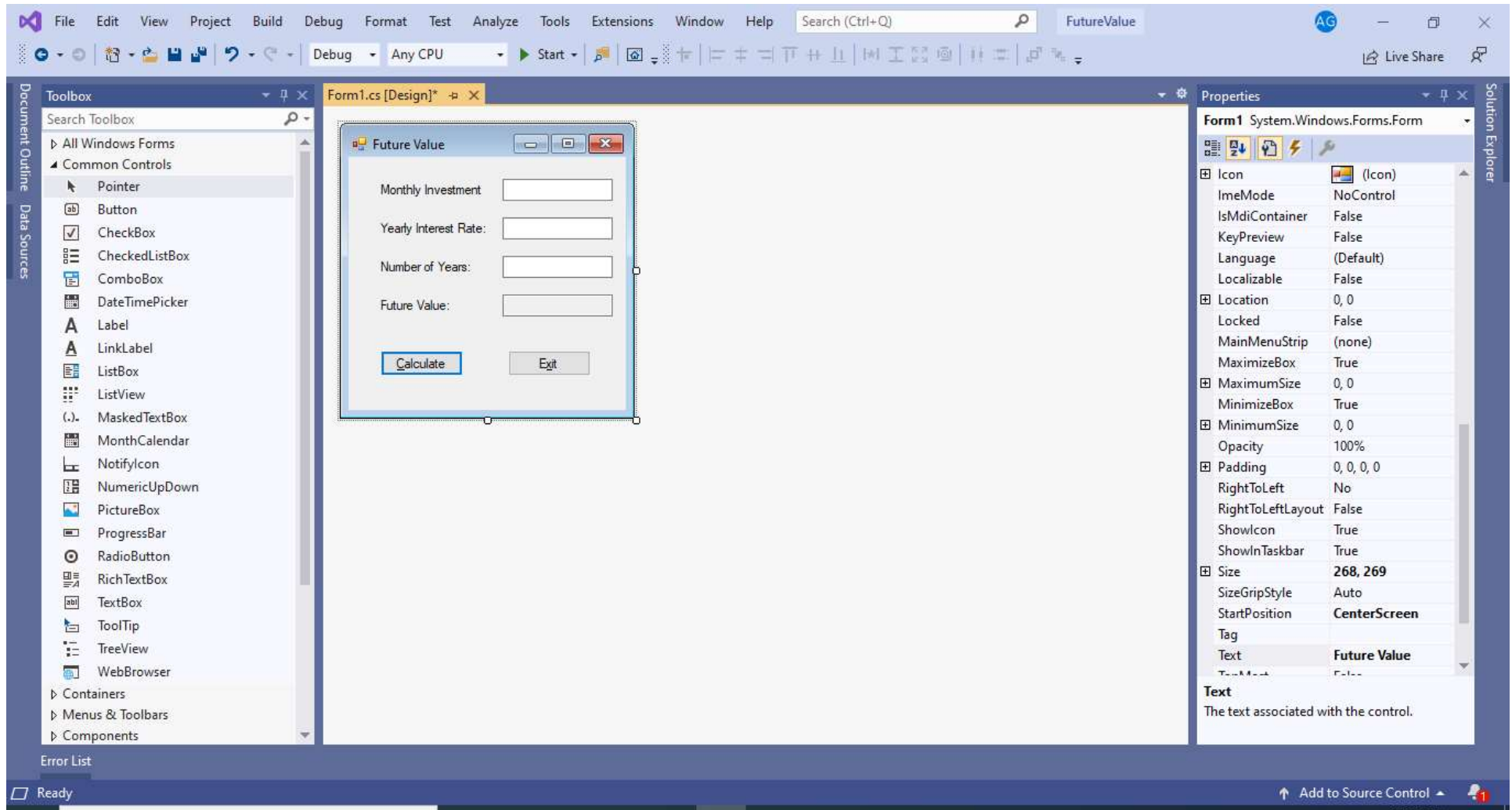


```
41 {
42     if (subtotal < 500)
43         discountPercent = .4m;
44     else if (subtotal >= 500)
45         discountPercent = .25m;
46 }
47 else
48 {
49     discountPercent = .1m;
50 }
51 switch (customerType)
52 {
53     case "R":
54         if (subtotal < 100)
55             discountPercent = .0m;
56         else if (subtotal >= 250 && subtotal < 500)
57             discountPercent = .25m;
58         else if (subtotal >= 500)
59             discountPercent = .3m;
60         break;
61     case "C":
62         if (subtotal != 0)
63             discountPercent = .2m;
64         break;
65     case "T":
66         if (subtotal < 500)
67             discountPercent = .4m;
68         else if (subtotal >= 500)
69             discountPercent = .25m;
70         break;
71     default: //if customer type isnt R,C OR T
72         discountPercent = .1m;
73         break;
74 }
```

Customer type	Subtotal	Discount percent	Discount amount	Total
R	300	25.0%	\$75.00	\$225.00
R	600	30.0%	\$180.00	\$420.00
C	600	20.0%	\$120.00	\$480.00
T	600	25.0%	\$150.00	\$450.00

## Ex. 5-2 Develop the Future Value application (Windows Forms project)

Develop the form, write the code and test the application



File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) FutureValue

Form1.cs [Design] FutureValue.FutureValue.Form1 btnCalculate\_Click(object sender, EventArgs e)

```
24 /*****
25  * Andrea Griffis
26  * Ex 5-2: Develop the Future Value Application
27  * 4/25/20
28  *****/
29 1 reference
30 private void btnCalculate_Click(object sender, EventArgs e)
31 {
32     //Getting inputs
33     decimal monthlyInvestment = Convert.ToDecimal(txtMonthlyInvestment.Text);
34     decimal yearlyInterestRate = Convert.ToDecimal(txtInterestRate.Text);
35     int years = Convert.ToInt32(txtYears.Text);
36
37     //Conversion to the same time period, from year to months
38     int months = years * 12;
39     decimal monthlyInterestRate = yearlyInterestRate / 12 / 100;
40
41     //Starting the loop with future value starts at zero
42     decimal futureValue = 0m;
43     for (int i = 0; i < months; i++)
44     {
45         futureValue = (futureValue + monthlyInvestment)
46             * (1 + monthlyInterestRate);
47     }
48     txtFutureValue.Text = futureValue.ToString("c");
49     txtMonthlyInvestment.Focus();
50 }
51
52 1 reference
53 private void btnExit_Click(object sender, EventArgs e)
54 {
55     this.Close();
56 }
```

Future Value

Monthly Investment: 100

Yearly Interest Rate: 12

Number of Years: 1

Future Value: \$1,280.93

Calculate Exit

100% No issues found Ln: 28 Ch: 63 SPC CRLF

Error List Locals Watch 1 Call Stack Exception Settings Immediate Window

Ready Add to Source Control



## Set breakpoints and step through the loops

The screenshot shows the Visual Studio IDE with a C# project named 'FutureValue'. The code in 'Form1.cs' is as follows:

```
31 //Getting inputs
32 decimal monthlyInvestment = Convert.ToDecimal(txtMonthlyInvestment.Text);
33 decimal yearlyInterestRate = Convert.ToDecimal(txtInterestRate.Text);
34 int years = Convert.ToInt32(txtYears.Text);
35
36 //Conversion to the same time period, from year to months
37 int months = years * 12;
38 decimal monthlyInterestRate = yearlyInterestRate / 12 / 100;
39
40 //Starting the loop with future value starts at zero
41 decimal futureValue = 0m;
42 for (int i = 0; i < months; i++)
43 {
44     futureValue = (futureValue + monthlyInvestment)
45                 * (1 + monthlyInterestRate);
46 }
47
48 txtFutureValue.Text = futureValue.ToString("c");
49 txtMonthlyInvestment.Focus();
50 }
51
```

A breakpoint is set at line 42. The 'Locals' window is open, showing the following variables and their values:

Name	Value	Type
this	{FutureValue.Form1, Text: Future Value}	FutureValue.Form1
sender	{Text = "&Calculate"}	object {System.Windows.Forms.Button}
e	{X = 38 Y = 20 Button = Left}	System.EventArgs {System.Windows.F...
monthlyInvestment	100	decimal
yearlyInterestRate	12	decimal
years	1	int
months	12	int

The status bar at the bottom indicates 'Ready' and 'Add to Source Control'.

Visual Studio interface showing the development of a Windows application named "FutureValue".

**Code Editor (Form1.cs):**

```
38 decimal monthlyInterestRate = yearlyInterestRate / 12 / 100;
39
40 //Starting the loop with future value starts at zero
41 decimal futureValue = 0m;
42 for (int i = 0; i < months; i++)
43 {
44     futureValue = (futureValue + monthlyInvestment)
45     * (1 + monthlyInterestRate);
46 }
47
48 txtFutureValue.Text = futureValue.ToString("c");
49 txtMonthlyInvestment.Focus();
50 }
51
52 1 reference
53 private void btnExit_Click(object sender, EventArgs e)
54 {
55 }
```

**Future Value Dialog Box:**

- Monthly Investment: 100
- Yearly Interest Rate: 12
- Number of Years: 1
- Future Value: (empty)
- Buttons: Calculate, Exit

**Locals Window:**

Name	Value	Type
------	-------	------

**Bottom Bar:**

- Error List
- Locals
- Watch 1
- Call Stack
- Exception Settings
- Immediate Window
- Loading symbols for FutureValue.exe
- Add to Source Control

File Edit View Project Build Debug Test Analyze Tools Extensions Window Help Search (Ctrl+Q) FutureValue

Debug Any CPU Continue

Form1.cs [Design] FutureValue FutureValue.Form1 btnCalculate\_Click(object sender, EventArgs e)

```
38 decimal monthlyInterestRate = yearlyInterestRate / 12 / 100;
39
40 //Starting the loop with future value starts at zero
41 decimal futureValue = 0m;
42 for (int i = 0; i < months; i++)
43 {
44     futureValue = (futureValue + monthlyInvestment)
45                 * (1 + monthlyInterestRate);
46 }
47
48 txtFutureValue.Text = futureValue.ToString("c");
49 txtMonthlyInvestment.Focus();
50 }
51
52 1 reference
53 private void btnExit_Click(object sender, EventArgs e)
```

100% No issues found Ln: 44 Ch: 17 SPC CRLF

Locals Search (Ctrl+E) Search Depth: 3

Name	Value	Type
▶ this	{FutureValue.Form1, Text: Future Value}	FutureValue.Form1
▶ sender	{Text = "&Calculate"}	object {System.Windows.Forms.Button}
▶ e	{X = 34 Y = 18 Button = Left}	System.EventArgs {System.Windows.F...
▶ monthlyInvestment	100	decimal
▶ yearlyInterestRate	12	decimal
▶ years	1	int
▶ months	12	int
▶ monthlyInterestRate	0.01	decimal
▶ futureValue	410.10050100	decimal
▶ i	4	int

Error List Locals Watch 1

Call Stack Exception Settings Immediate Window

Ready Add to Source Control